ISSN: 2277-3754
ISO 9001:2008 Certified
International Journal of Engineering and Innovative Technology (IJEIT)
Volume 3, Issue 1, July 2013

Profiling E-Governance Users using Biclustering

S.Chakraverty, G.Rani, A.Bardhan
Netaji Subhash Institute of Technology, Delhi University, Delhi, India

Abstract- With the widespread and proactive participation of citizens through various e-governance applications, democracy in the modern era has acquired an entirely new dimension The shear diversity of e-governance users has spurred on a fresh interest in designing adaptable e-governance systems. The first step in meeting this challenge is to develop a fast and versatile automated technique to categorize users on the basis of a similarity in their online identities and behaviors. In this paper we employ a modified version of the Cheng and Church Biclustering algorithm, hitherto used primarily in the field of Genetics, to extend its applicability to a classification of e-governance users. Taking a different route from conventional approaches, we tap a variety of dynamically varying parameters that characterize the online behavior of users with a view to improving the cohesiveness of user clusters. These include the navigation patterns of a user, her access frequency and the interactivity level during her web experience. We adopt two strategies for clustering. A single level strategy categorizes users on all the three parameters taken together using the Cheng and Church (CC) algorithm. We also employ a two level clustering strategy that first finds biclusters on the basis of individual parameters using CC and then the uses cluster ids to classify the users at a second level by and K-Means clustering. An analysis of the granularity of clusters and execution time for different strategies and datasets reveals that the single level strategy is useful in categorizing experienced users who have attained a degree of familiarity with the portal and are able to change their behavior frequently. Such a group of users is quite variegated. On the other hand, the two level strategy provides a better way to classify beginners who show very slow changes and are more uniform in their web interactions.

***Index Terms*— Biclustering, , e-governance, single levels strategy and two level strategy.**

I. INTRODUCTION

E-governance (e-gov) is an admirable initiative by governments across the globe to provide a transparent and efficient way for public to utilize their services online [1] [2]. The sheer variety of services available and the wide heterogeneity of data presented at e-gov portals restrict their use to only those people who are skilled web users. To increase the availability of e-gov services to all kinds of users we need to develop adaptable e-gov systems which can re-orient user interfaces, fine-tune data presentation and provide tailor-made recommendations for browsing, in a manner that is most suitable and efficient for different categories of users. Such facilities can greatly reduce the digital divide that has become the bane for modern society.

The first step towards development of user oriented interface systems is to develop an automated mechanism to derive knowledge about users' online usage patterns. This requires systematically profiling all users who are engaged in accessing e-gov portals for one or more purposes and collating the data to extract common browsing behavioral characteristics based on which users can be classified into distinct categories.

In this paper, we propose the most appropriate clustering strategies for differently evolving user bases. The users' online behavior is captured with a comprehensive set of attributes including navigation path, frequency of accessing each page and the interactivity level sustained by users during their web interaction. We present a comparative analysis of single level and two level clustering strategies by employing three techniques: Cheng and Church biclustering, K-Means clustering and a combination of the two. Our experiments investigate the impact on execution time, quality and number of clusters when the base dataset of users' web behaviors is changed with minor modifications and major modifications.

The remaining part of the paper is organized as follows. Section 2 outlines related prior work. Section 3 presents the proposed algorithm for user categorization using the Biclustering technique. In section 4 we evaluate experimental results using a dataset of online users. In Section 5 we present a comparative evaluation of our work. We conclude and provide ideas on future extensions in section 6.

II. PRIOR WORK

A review of existing literature demonstrates the extensive use of web mining techniques to categorize online users according to their behavioral aspects [3].

A comparison/ survey of biclustering techniques is given in [4] [6] and [9]. The biclustering algorithm used in [7] uses a greedy approach for categorizing data items to identify different gene groups. A set of genes constitute one dimension of a bicluster matrix while their corresponding characteristics form another. This method discovers one bicluster per iteration and executes repeatedly to find more than one biclusters. The authors in [7] defines a threshold value for the *Mean Square Residue (MSR)* which is the optimum value that achieves maximum cohesiveness among the elements of all biclusters. Discovering coherent biclusters from a dataset matrix is a significant research contribution that has found wide application in finding two dimensional clusters from data. However, the work in [7] ignores the possibility of fine tuning of value of threshold to improve the cohesiveness of biclusters. Also, the time complexity for extracting a single bicluster is high which remains same till all the biclusters are discovered.

In [5], the authors have applied the Biclustering technique for classifying online users in virtual campuses according to their online behavior. The authors employ CC algorithm to categorize online users on the basis of their navigation

ISSN: 2277-3754
ISO 9001:2008 Certified
International Journal of Engineering and Innovative Technology (IJEIT)
Volume 3, Issue 1, July 2013

pattern, time of access and location of access on a website. The extension in scope of biclustering from genetics to an online application is an innovative idea. But the authors consider only basic navigation pattern as an attribute. In contrast, we take into account other important parameters such as interactivity level and frequency of access that describe online behavior.

III. PROPOSED WORK

A. Definitions

An e-gov portal, *P* is defined as a collection of n interlinked web pages; $\{W_i\}$. An e-gov user, e_i accesses a sequence of pages. We define four levels of page access. The base level: $Level_0$ indicates nil user activity on a page for a time slot of at least one minute. The first level: $Level_1$ includes an active stay on the same page. The second level: $Level_2$ shows traversals to different page(s) of same website. The third level: $Level_3$ is registered when the user switches to a page of another website.

(i)Session: It represents the interval between once log in and log out. In a session of 30 minutes, navigation behavior of a user is recorded every minute. This splits the session duration into 30 equal slots of one minute each.

(ii)*The navigation pattern, NP* is the sequence of switches from one page level to another which a user does every minute in one session. We assume that a user starts her session with an access to the home page. This makes 1 as the first digit of each navigation pattern. The next pattern digit remains 1 if the user remains active on same page. The pattern digit becomes 0 if she does not navigate. It becomes 2 when a user switches to another page of same website. It records 3 if she navigates to another website. It is a sequence of 4 integer values viz 0,1, 2 and 3 in different orders to make a complete sequence of length 30. For eg. 1200000003 be a fragment of NP that records behavior of a user for a time duration of 10 minutes. In this pattern the user accesses home page of a website in first minute, next minute she switches to another page of same website. She sticks to the same page for next seven minutes. At the end, she navigates to another website. We introduced access frequency and interactivity level in [11] for developing user oriented dynamic recommendation system. We utilize these parameters for user clustering.

(iii) Access frequency, AF: It is the number of times a user switches from a specific page to another on the e-gov portal in one session. It is a sequence of N integer numbers. For eg. If N=8, Let 5 3 1 4 1 1 1 1 be the access frequency of a user in one session. It indicates that a user 5 times switches to page one, 3 times to page 2, 1 time to page three and 4 times to page four. She navigates to pages five, six, seven and eight only once during one complete session.

(iv) Interactivity Level, IL: Interactivity is the bidirectional interaction between a user and the system. We designate online form filling(OF), blogs(B), feedback(FB) and chat(C) options available on a website as elements that decide the interactivity level of a user. These elements facilitate a two way communication of users with the system. The interactivity level they offer are chosen as: $IL_{OF}=1$, $IL_B=2$, $IL_C=3$ *and* $IL_{FB}=4$.

Using the interactivity level of each element, IL_x and time duration, $\Delta T_{i,x}$ for which a user, i accesses a particular element x, we define the interactivity level, *IL* of a user *i* as:

$$IL_i = \frac{\sum_x (IL_x \bullet \Delta T_{i,x})}{\sum_x \Delta T_{i,x}} : x = \{OF, B, C, FB\}$$

IL is a float value. We scale this to integer value so as to maintain homogeneity of its data type with NP and AF, using the algorithm scaling_IL shown in figure3.1.

The algorithm takes an array of real valued interactivity level of users. After initializing range and a counter, it enters the difference calculation phase. In this phase, it searches for rows/columns of maximum (Step 2.1 below) and minimum value, Step 2.2 below) of *IL*. Now, it calculates the difference of these two values as shown in step 2.3. In next phase, in step 3.1.2 it iteratively converts each real value of *IL* to integers.

Algorithm **_Scaling_IL** (
Input: A = {A_1, A_2, A_3 ..., A_n} An array of 'n' real numbers
Output: B = {B_1, B_2, B_3 .. , B_n} An array of 'n' integers)
1. Initialization phase:
1.1. Range = 10
1.2. i = 0;
2. Difference Calculation Phase:
2.1. Find max real number, max_{real} in A
2.2 Find min real number, min_{real} in A
2.3. *diff = max_{real} - min_{real}*
3. Iteration phase:
3.1 **while** i < n **do**
3.1.1. j = 1
3.1.2. **while** j < Range **do**
3.1.2.1 **if** A_i >= *min + (j - 1)*(diff/Range)* **and** A_i <=*(min + j) *(dif/Range)* **then do**
B_i = j
else do
j = j + 1
end if
end while
3.1.3. i = i + 1
end while
return B;

Fig.1: Algorithm for Scaling Interactivity level

(v)Coherence: Mean Square Residue, MSR is the measure of coherence. It is the metric for measuring the quality of a bicluster. The lowest value *of MSR* indicates the highest coherence among elements of input data matrix.

(vi)Threshold,δ: This is the optimum value of *MSR* beyond which the data items become non coherent. Its value greatly varies with versatility in data set, exemplified as Cheng and Church use δ=300 to get biclusters from data set of yeast. They use δ=1200 for human data set. Here, we fine tune the value of threshold for our dataset from 0 to 1. In the e-gov

ISSN: 2277-3754
ISO 9001:2008 Certified
International Journal of Engineering and Innovative Technology (IJEIT)
Volume 3, Issue 1, July 2013

application 0 threshold value is most suitable because it requires users with similar web behavior together in a bicluster. This adds to develop a common interface for similar users.

B. Biclustering Algorithm

We propose an algorithm which finds groups of users having common interests towards e-gov services. The algorithm is founded upon the CC Biclustering algorithm [7]. We implement a two level CC algorithm. In addition, we contribute in restricting the execution of addition phase in data set of e-gov application. In this data set, addition phase is invoked occasionally when the deletion phase give an empty matrix as result.

The algorithm Biclustering given in figure3.2 finds one bicluster from the given matrix. The algorithm is invoked iteratively to get more biclusters.

Input Matrix: A matrix X=[E][B] is a set of rows [E] and a set of columns[B]. Rows represent e-gov users and columns store browsing behavioral attributes of the corresponding users. Thus, E={E1,E2…En } and B={B1, B2…Bn }.

Let M be a set of sub matrices of X. A factor called Mean Square Residue (MSR) of a given matrix is calculated as explained below. For a given threshold parameter δ for MSR and a sub-matrix Mi, the objective of the algorithm is to ensure that δ > MSR (Mi). After performing Initialization tasks, the algorithm alternates between Reduction and Extension phases. These phases are now described.

1 .Initialization Phase: This phase initializes a set [E] for rows and a set [B] for columns of the input matrix. We initialize the X itself as an initial (largest) bicluster. This contains all the rows and columns of input data matrix. We set a flag value to zero. This flag value indicates the switching of algorithm between reduction and extension phases.

2. Calculating Residues: This function defines the Residue, R(xi,j) for each element, xi,j. of the input matrix X. It uses the mean of ithrow over all the columns: $\bar{x}_{iJ}$; the mean of jth column over all the rows: $\bar{x}_{Ij}$ and the mean of whole matrix: $\bar{x}_{IJ}$.

$$R(x_{ij}) = x_{ij} - \bar{x}_{iJ} - \bar{x}_{Ij} - \bar{x}_{IJ}$$

The Residue is stored in a separate matrix, R.

3. Calculating Mean Square Residue (MSR): Using the number of rows |E|, the number of columns |B| and the Residue, $R(x_{i,j})$ The *Mean Square Residue* of matrix, X is.

$$MSR(E,B) = \frac{\sum_{i\in I, j\in J} (R(x_{ij}))^2}{|E||B|}$$

***4. Reduction Phase*:** This phase performs following computations iteratively till the *Mean Square Residue, MSR* exceeds the threshold, δ and the matrix does not become empty. Flag remains zero in this phase.

Row Mean Residue: Mean of i[th] row of Residue matrix is calculated as:

$$MRiJ = \frac{\sum_{j\in J} R(rij)}{|J|}$$

Column Mean Residue: Mean of j[th] column of Residue matrix is computed as:

$$MRIj = \frac{\sum_{i\in I} R(r_{ij})}{|I|}$$

Comparing Mean Residue: We find those row(s) or column(s) from the Residue matrix which have the highest value of their mean(s). This uses the following formula[4.1.3, figure3.2]:

$$MR_{\max} = \max(\max(MRi_J \forall j \in J), \max(MR_{Ij} \forall i \in I))$$

Now, this phase deletes those rows and columns from the input matrix which have maximum, max value of their mean Residue. High Mean Residue reduces the coherence among dataset elements as stated in[7]. Deletion of multiple rows occurs when more than one row have same MR_{iJ} . It removes multiple columns if more than one columns have same max value of MR_{Ij} . These rows and columns are marked as non coherent. It brings highly cohesive data together in a single bicluster.

5. Extension Phase: Our algorithm executes this phase when the value of Residue of all users is same. In such a case, reduction phase deletes all the rows which results in an empty matrix. This phase continues till the threshold remains greater than *MSR*. Flag remains one throughout this phase. Refraining the extension phase from executing in each iteration, reduces the time required for finding all the biclusters. In this phase, our algorithm calculates the Mean Residue, of each row, MR_{iJ} and Mean Residue of each column, MR_{Ij} for the matrix which has been deleted in reduction phase. For making these computations, it uses the *Row Mean Residue*, $MRiJ$ *(step 4.2)* in figure 3.2 and *Column Mean Residue*, $MRIj$ formulae defined in reduction phase (step 4.3) Now, it finds the row(s) or column(s) having minimum, min value of Mean Residue using the formula :

$$MR_{\min} = \min(\min(MR_{iJ} \forall j \in J), \min(MR_{Ij} \forall i \in I))$$

This phase adds those row(s) or column(s) to the matrix which have minimum, min value of Mean Residue. Algorithm to find Biclusters of E-Gov users been shown in figure 3.2

Algorithm_Biclustering (
Input: Matrix X[E][B], threshold δ;)
Output**:** Bi-cluster matrix Bm [row][col] ;

1. **Initialization phase:**
   1.1 Set row = |E|, col = |B|, Bm = X; flag = 0;
   1.2 Create an empty residue matrix R[E][B].
2. **Residue Calculation:**
   Calculate residue for each element in X and store in R
   $$R(x_{ij}) = x_{ij} + \bar{x}_{iB} - \bar{x}_{Ej} - \bar{x}_{EB};$$
3. **Mean Square Residue computation:**
   $$MSR(E,B) = \frac{\sum_{i\in E, j\in B} (R(x_{ij}))^2}{|E||B|}$$

ISSN: 2277-3754
ISO 9001:2008 Certified
International Journal of Engineering and Innovative Technology (IJEIT)
Volume 3, Issue 1, July 2013

;

4. **Reduction phase:**

4.1 **while ((** MSR(E,B) > δ) and (flag == 0)) **do**
Calculate MR_{iJ} for all rows and columns (Eq 3and 4)

4.2 Find maximum MR value:

$$MR_{\max} = \max(\max_{i \in E}(MR_{iB}), \max_{j \in B}(MR_{Ej}))$$

4.3 **Delete** all rows or columns with maximum MR of MR_{iJ} or MR_{Ij} .

4.4 **if** (row==0 OR col ==0) **then** Set flag=1

End while

5. **Extension phase:**

5.1 **while ((** MSR(E,B) < δ) and (flag==1)) **do**
Calculate MR_{iJ} for all rows and columns (Eq 3and 4)

5.2 Find minimum MR value:

$$MR_{\min} = \min(\min_{i \in E}(MR_{iB}), \min_{j \in B}(MR_{Ej}))$$

5.3 Add multiple rows or columns with min MR

end while //

Fig 2 Algorithm to find biclusters

IV. EXPERIMENTAL RESULTS

A. *Generating Data Sets:*

The system generates input datasets of four sizes comprising 50,100,150 and 200 users respectively, using a randomization function. We made three cases for each of the datasets.

(i) Dataset 1- Base dataset: Each dataset contains five distinct groups. Users in a group are same in their web behavior. They completely differ in their web behavior from users of other groups. The groups have their unique *NPs*. They also have distinct *AF* and *IL* values. Behavioral attributes decide the number of columns. There are 39 columns in each matrix. Initial thirty columns indicate *NP*. Next eight columns represent *AF*. The last column shows the *IL* values. Table 4.1 shows the base dataset for 12 users. In this sample dataset, it takes a fragment of size 10 of *NP*, complete *AF* of size 8 and *IL* of size 1. We can observe that the first user shows the same web behavior as the 10th user. The 2nd user is same as the 11th user in her web behavior, and so on.

(ii) Dataset 2 - Base Dataset with Minor perturbations: Minor modifications are important to show the behavior of beginners. They change their behavior at a slow rate. Table 4.2 shows the base dataset with minor perturbations of 12 users. We perform minor modifications in the datasets 1. Number of groups in dataset 2 are twice the number of groups in dataset 1. It makes changes at the same positions in the data of 50% users in each group. It changes 3-6 digits in 30 digit long NP, 2 digits in 8 digit long *AF* and add 0.25 in the value of *IL*. We thus expect a variation in number of clusters from 5 to 10. The data set gives 5 clusters if the algorithm Biclustering given in fig 3.2 completely ignores minor modifications. In this condition, users with minor modifications are categorized to single cluster on the basis of similarity among data elements. It provides 10 clusters if it considers minor modifications which makes a distinction among data elements on the basis of minor modifications.

(iii) Dataset 3 - Dataset 2 with Major Perturbations: Major modifications are significant as they show behavior of experienced users. These users frequently change their behavior while accessing a web portal. Table 4.3 shows the data of 12 users with major changes. The data generation system mutates the dataset with major changes. It applies these changes in those 50% users of each group who have already been altered with minor modifications. Here, it changes 9-15 digits in 30 digit long *NP*, 5 digits in 8 digit long *AF* and add 0.5 in *IL*. We expect demarcation of data elements among 5 to15 clusters. If any strategy yields 5 clusters, it shows complete ignorance of the modifications. Generation of 15 clusters indicates the due consideration of the modifications which distinguishes the data elements completely.

B. *Experimental Strategies:*

We follow two strategies which are useful in recording the web behavior of different e-gov users.

***(i) Single Level Strategy*:** Here, we record *NPs* of all the users, for a session of 30 minutes, in a single fragment of size 30. We store all the three parameters: *NP*, *AF* and *IL* in a single matrix. The number of rows in the matrix varies with the change in number of users. Number of columns remains 39 in each case as the number of parameters is constant for each data size.

(ii) *Two level Strategy:* We divide *NP* which has been generated in a session of 30 minutes, in three fragments of equal sizes. Each fragment displays the *NP* that has been recorded for time duration of 10 minutes. We represent these fragments as NP_1, NP_2 and NP_3. Each fragment consists of a sequence of 10 integer numbers. Breaking of *NP* into three different fragments in the two-level strategy assists in finding users who are partially similar in their navigation behavior. It represents the behavior of those users also who access the portal only for a short duration of 10 minutes in a single session. Such users log out the web portal before completing the session of 30 minutes. Now, we store NP_1,NP_2, NP_3, *AF* and *IL* in five different matrices of different sizes. For 100 users, exemplified as $NP_1[100][10]$, $NP_2[100][10]$, $NP_3[100][10]$, AF[100][8] and IL[100][1].

First three matrices have user IDs to represent rows while page IDs for columns. Fourth matrix contains the *AF* of 100 users. User IDs denotes the rows and frequency of switching from one page level to another represents the columns. Fifth matrix displays the *IL* of each user. Its rows contain user IDs while columns include the *IL* of corresponding users.

Each strategy has its own significance. Single level strategy classifies the experienced users more accurately as they change their pattern frequently. Two level strategy is significant in categorizing naïve users as they navigate less frequently.

ISSN: 2277-3754
ISO 9001:2008 Certified
International Journal of Engineering and Innovative Technology (IJEIT)
Volume 3, Issue 1, July 2013

C. Clustering Techniques:

We performed experiments on each of the four datasets. For each dataset we used the following techniques:

(i) *Single-level and two-level Cheng and Church Biclustering algorithms:* In single level strategy, We apply CC algorithm on input matrices of [50][39],[100][39],[150][39] and [200][39] individually. It finds biclusters of users on the basis of all the three parameters collectively. In two level strategy, it follows the same procedure as in two level differentiated approach explained below in (iii) except that it applies CC biclustering algorithm at both the levels instead of using CC biclustering at the first level and K-Means clustering at the second level.

We judge the effect of increase in dataset size on the time of executing the algorithm Biclustering shown in fig 3.2. It also gives a chance to compare the execution time needed for categorizing three different data sets for each size. One dataset represents the users without any modification, second with minor modifications and third with combination of minor and major alterations.

(ii) Single-level and two-level K-Means clustering technique: This works in a similar manner above as except that K-means clustering is applied instead CC biclustering is used.

(iii) Two-level differentiated approach: Cheng and Church bilcustering algorithm was applied at the first level of a two level strategy and K-Means was applied at the second level. At the first level, we apply Cheng and Church algorithm on initial four matrices individually. This is useful in bringing similar users together. The sizes of matrices which store NP_1, NP_2 *and* NP_3 are [Number of users][10]. The size for *AF* matrix is [Number of users][8]. We use K-Means clustering technique on the fifth matrix which holds the values of *IL*. It finds clusters on the basis of single valued *IL* in lesser time than using CC algorithm.

First level clustering assigns cluster IDs to users on the basis of each of the selected parameters, individually. Users in one bicluster get same cluster ID on the basis of one parameter. They may get different cluster IDs on the basis of different parameters. We assign cluster IDs at the sake of number of iteration when it becomes a cluster.

At second level, we apply K-means clustering algorithm on the matrix which has user IDs at rows and cluster IDs at columns. We are able to do so, because similar users have same cluster IDs. K-Means bring them together in the same cluster. The size of matrix is [Number of users][5]. Each column represents cluster IDs on the basis of parameters such as NP_1, NP_2, NP_3, *AF* and *IL*. This gives the groups of those users who are similar at both the levels.

Results give a comparative evaluation of quality of clusters. It also provides a comparison in the time that each algorithm takes for finding clusters from each case of data set for different data sizes using these two techniques. It gives an analysis of number of clusters in each case.

V. Results Analysis and Discussion

We evaluate our algorithm on the basis of following measures:

(i) Quality of Biclusters: Average Mean Square Residue *MSR*, is a measure of the quality of a bicluster. Minimum value of *MSR* indicates the best quality of a bicluster. This shows maximum coherence among the elements of a bicluster. Experimental results on each data size show that *MSR* is zero for all the clusters that we obtained using different clustering techniques for both single level as well as two level strategies. This is due to the fact that in e-gov application, δ is kept 0 in order to bring only those users together who are navigating in the same way on a portal.

(ii) Number of Biclusters: The number of clusters obtained using single level approach matches the number of clusters that we expect in the data set. Figure 1 demonstrates that both K-Means clustering as well as CC biclustering algorithms give the same number of clusters for the base data set and data set with minor modifications. In the case of the base data set that was perturbed with minor as well as major modifications, K-Means gave the expected number of clusters. Here, CC biclustering algorithm fuses two or more clusters on the basis of similarity among data elements. This reduces the number of biclusters. Thus, we conclude that both K-Means and CC algorithms keep the users with same web behavior in the same cluster.

Figure 2 shows that K-Means clustering finds the expected number of clusters in the two level approach also. Two-level CC biclustering algorithm segregates the users in least number of clusters. This occurs as it brings those rows and columns together which have zero value of residue. It does so before calculating the value of MSR. This results in grouping of dissimilar users in a group.

In the two-level differentiated strategy, the CC algorithm applied at the first level finds clusters of users who are partially similar in their web behavior. They show similarity in any subset of the individual attribute NP_1, NP_2, NP_3, *AF* or *IL*. Now applying K-Means clustering at second level using the cluster IDs that we obtained as a result of first level clustering, bring those users together who have same cluster IDs at first level. This strategy gives more number of clusters as compared with the previous two strategies.

Thus, we conclude that a combination of CC biclustering algorithm at level first and K-Means clustering at second level is the most suitable approach for categorizing e-gov users as it clusters similar users together. It considers not only the partial similarity among users but is also useful in categorizing those users who access the portal for a short duration.

Further, variations in the dataset also contribute to the increase in number of clusters generated. Not only major, even minor modifications in the dataset are sufficient to make a separate category.

ISSN: 2277-3754
ISO 9001:2008 Certified
International Journal of Engineering and Innovative Technology (IJEIT)
Volume 3, Issue 1, July 2013

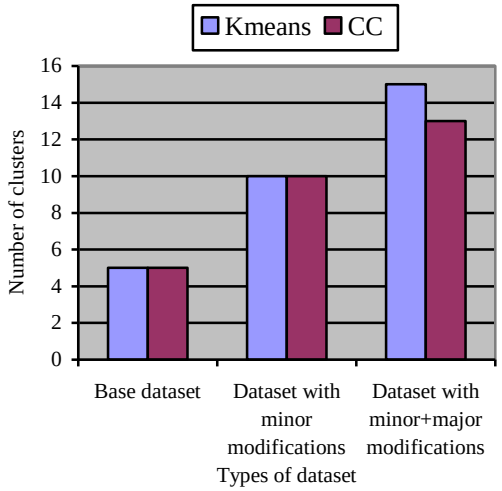

Fig. 1, Clusters of 200 users using single level strategy

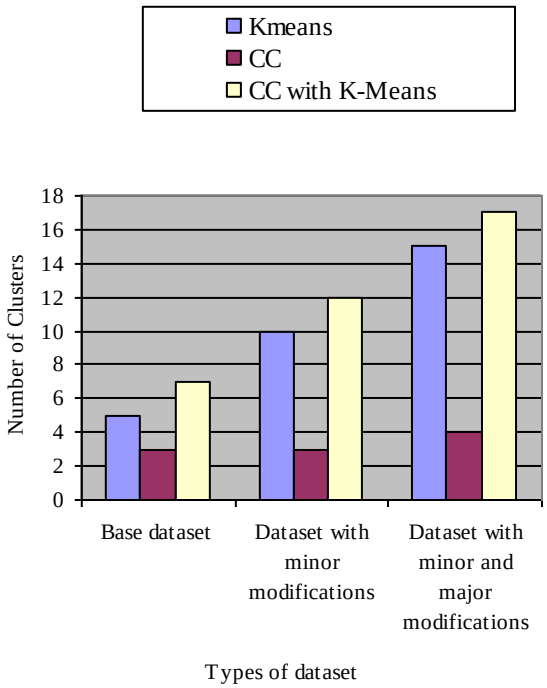

Fig 2, Clusters of 200 users using two level strategy

***(iiI) Execution time*:** Figure 3 implies that in two level strategy CC biclustering algorithm takes the maximum time. K-Means clustering finds the clusters in minimum time. Applying CC biclustering algorithm at first level and K-Means clustering at second level binds the similar users together taking lesser time than CC biclustering algorithm.

Figure 4 clearly shows that in single level approach, CC biclustering algorithm takes quite longer time than K-Means clustering on the same data size. The figure depicts that categorizing the base data set takes minimum time. Time increases with increase in modifications in the base dataset. This is because of increase in number of clusters on modifying the base dataset. There is an increment in execution time of each algorithm when we modify the base dataset. Minor modification shows lesser time increment than major modifications. This is because of increase in number of clusters on modifications.

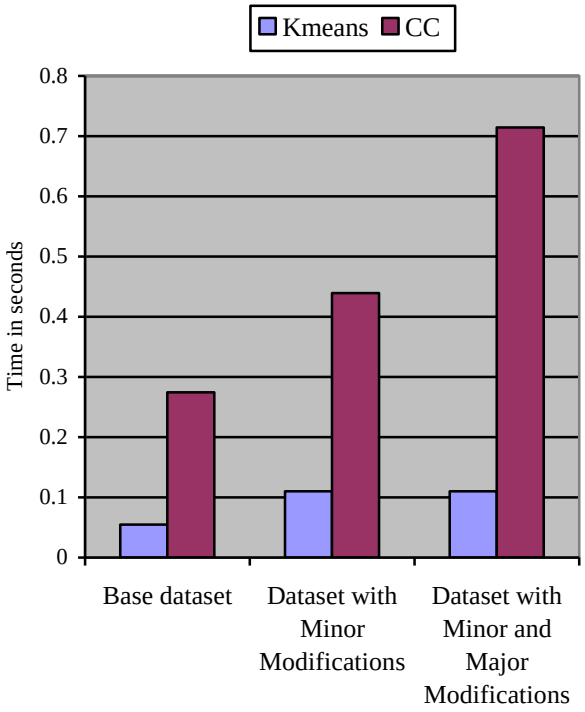

Fig 3, Clustering of 200 users using single level strategy

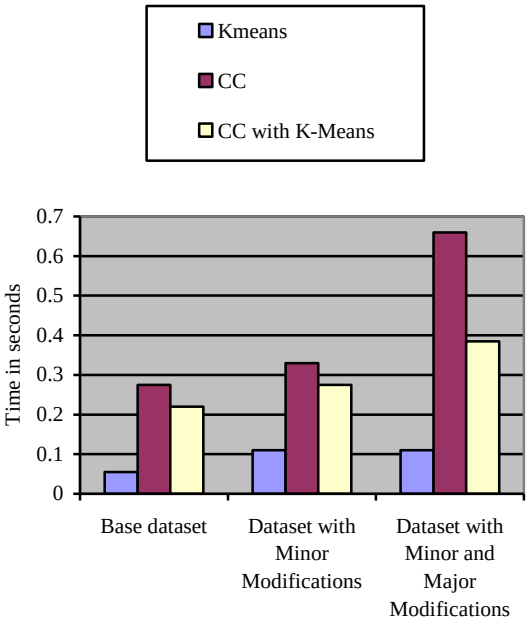

Fig 4, Clustering of 200 users using Two level strategy

ISSN: 2277-3754
ISO 9001:2008 Certified
International Journal of Engineering and Innovative Technology (IJEIT)
Volume 3, Issue 1, July 2013

VI. CONCLUSION AND FUTURE WORK

This paper presents a two level biclustering approach which responds intelligently to dataset matrices. It uses greedy approach to keep similar e-gov users together in a single bicluster. It categorizes users on the basis of a variegated set of attributes such as *NP*, *AF* and *IL*. Finally it categorizes users on the basis of the combined attributes.

Experimental results reveal that two level strategies in which we apply CC algorithm with conditional execution of addition phase at first level and K-Means clustering at second level is more effective in separating dissimilar users than one level approach for classifying e-gov users using CC biclustering algorithm. It is also more effective than using CC algorithm at both the levels in the two-level approach. This brings together not only same users but also those users who are partially similar in their web behavior.

Figure-4 clearly shows the comparison in time of execution of K-Means clustering and CC algorithm in both the strategies. It depicts that executing K-Means clustering has significantly lower time of 0.1 seconds than CC biclustering algorithm in single level strategy, which takes 0.7 seconds for the dataset with minor +major modifications at the dataset size of 200 users.

Figure 3 presents that, in two level strategy using CC biclustering at level1 and K-Means clustering at level 2 is more efficient than using CC algorithm at both the levels on the same dataset as it completes its execution in 0.38 seconds which is sufficiently less than 0.65 seconds in case of CC algorithm.

This work illustrates the applicability of CC bi-clustering algorithm from the field of genetics to e-gov. In this case too, not only the similarity among users but also the similarity among behavioral attributes observed during a specified time interval comes into play to achieve a well balanced categorization.

In the present work, we have tested our algorithm on a four randomly generated datasets of 50,100,150 and 200 users. We are currently working towards developing an adaptable e-gov system using biclustering techniques.

REFERENCES

[1] Ratneshwar; A.K. Tripathi;, "Some Component Generation Approaches for E-Governance Systems", International Journal of Public Information system, IJPIS Vol.2, 2010, pp.133-147.

[2] Z.Ebrahim; Z.Irani;, "E-Government Adoption :Architecture and Barriers", Business process management Journal, Emerald Group Publishing House, Vol II, No.5,. pp. 589-611, 2005.

[3] K. Etminani; A.R Delui.; N.R. Yanehsari; M Rouhani; , "Web usage mining: Discovery of the users' navigational patterns using SOM," First International Conference on Networked Digital Technologies, 2009. NDT '09., pp.224-249, 28-31 July 2009.

[4] N.K Verma; S. Meena; , A. Singh; Yan Cui; S Bajpai; A. Nagrare; , "A comparison of biclustering algorithms", International Conference on Systems in Medicine and Biology (ICSMB), 2010 , pp.90-97, 16-18 Dec. 2010.

[5] F. Xhafa; S. Caballé; L. Barolli, A Molina; R. Miho; ,"Using Bi-clustering Algorithm for Analyzing Online Users Activity in a Virtual Campus," 2nd International Conference on Intelligent Networking and Collaborative Systems (INCOS), 2010, pp.214-221,24-26Nov.2010.

[6] S.C. Madeira; A.L. Oliveira; "Biclustering algorithms for biological data analysis: a survey," IEEE/ACM Transactions on Computational Biology and Bioinformatics, vol.1, No.1, pp.24-45, Jan.-March2004.

[7] Y.Cheng ; G.M.Church; " Biclustering Expression Data", Proc. 8th International Conference on Intelligent systems for Molecular Biology", ISMB'00, pp.93-103, 2000.

[8] J *Yang*, H. Wang, W. Wang, and P. Yu, "*Enhanced Biclustering on Expression Data*," Proc. Third IEEE Symp. Bioinformatics and Bioengineering (BIBE '03), pp.321-327, 2003.

[9] J.Yang; W.Wang; H.Wang ;P.Yu; "Clusters: Capturing Subspace Correlation in large data sets, Proc. 18th IEEE, International conference on Data Engineering, pp 517-528, 2002.

[10] G. Rani; S. Chakraverty; "Boosting Interactivity in E-Governance", International conference on communication languages and signal processing in reference to 4G technologies, ICCLSP 4G 2012, New Delhi, 24-25th January 2012.

[11] S.Chakraverty; G.Rani; D.Anand; B.Singla; "Experience based Recommendation System for E-Governance", International conference on e-Learning, e-Business, Enterprise Information Systems, and e-Government, EEE-12, Las Vegas,US. 16-19 July 2012..

ISSN: 2277-3754
ISO 9001:2008 Certified
International Journal of Engineering and Innovative Technology (IJEIT)
Volume 3, Issue 1, July 2013

APPENDIX

Table 4.1, Base dataset

UID	NP	NP	NP	NP	NP	NP	NP	NP	NP	NP	AF	AF	AF	AF	AF	AF	AF	IL
1	1	2	2	2	0	2	2	2	2	0	5	3	1	4	1	1	1	10
2	1	0	2	0	0	2	0	2	2	1	4	4	2	2	3	2	2	0	4
3	1	2	0	1	2	1	0	2	2	2	5	2	3	4	1	1	1	1	4
4	1	1	2	2	2	2	2	2	2	0	3	3	2	2	1	1	1	1	1
5	1	1	2	0	0	1	0	2	2	2	5	4	2	2	2	2	2	2	9
6	1	2	2	2	0	2	2	2	2	0	5	3	1	4	1	1	1	1	10
7	1	0	2	0	0	2	0	2	2	1	4	4	2	2	3	2	2	0	4
8	1	2	0	1	2	1	0	2	2	2	5	2	3	4	1	1	1	1	4
9	1	1	2	0	0	1	0	2	2	2	3	3	2	2	1	1	1	1	1
10	1	2	2	2	0	2	2	2	2	0	5	4	2	2	2	2	2	2	9
11	1	0	2	0	0	2	0	2	2	1	5	3	1	4	1	1	1	1	10
12	1	2	0	1	2	1	0	2	2	2	4	4	2	2	3	2	2	0	4

Table 4.2, Base dataset with Minor Perturbations

UID	NP	NP	NP	NP	NP	NP	NP	NP	NP	NP	AF	AF	AF	AF	AF	AF	AF	AF	IL
1	1	2	1	2	0	2	2	2	2	0	5	3	6	4	1	1	1	6	10
2	1	0	1	0	0	2	0	2	2	1	4	4	7	2	3	2	2	5	4
3	1	2	2	1	2	1	0	2	2	2	5	2	8	4	1	1	1	6	4
4	1	1	1	2	2	2	2	2	2	0	3	3	7	2	1	1	1	6	1
5	1	1	1	0	0	1	0	2	2	2	5	4	7	2	2	2	2	7	9
6	1	2	2	2	0	2	2	2	2	0	5	3	1	4	1	1	1	1	10
7	1	0	2	0	0	2	0	2	2	1	4	4	2	2	3	2	2	0	4
8	1	2	0	1	2	1	0	2	2	2	5	2	3	4	1	1	1	1	4
9	1	1	2	2	2	2	2	2	2	0	3	3	2	2	1	1	1	1	1
10	1	1	2	0	0	1	0	2	2	2	5	4	2	2	2	2	2	2	9
11	1	2	1	2	0	2	2	2	2	0	5	3	6	4	1	1	1	6	10
12	1	0	1	0	0	2	0	2	2	1	4	4	7	2	3	2	2	5	4

Table 4.3, Base dataset with Major Perturbations

UID	NP	NP	NP	NP	NP	NP	NP	NP	NP	NP	AF	AF	AF	AF	AF	AF	AF	AF	IL
1	1	0	1	2	0	0	2	2	2	1	5	3	6	4	1	1	1	6	10
2	1	1	1	0	0	0	0	2	2	2	4	4	7	2	3	2	2	5	4
3	1	0	2	1	2	2	0	2	2	0	5	2	8	4	1	1	1	6	4
4	1	2	1	2	2	0	2	2	2	1	3	3	7	2	1	1	1	6	1
5	1	2	1	0	0	2	0	2	2	0	5	4	7	2	2	2	2	7	9
6	1	2	2	2	0	2	2	2	2	0	5	3	1	4	1	1	1	1	10
7	1	0	2	0	0	2	0	2	2	1	4	4	2	2	3	2	2	0	4
8	1	2	0	1	2	1	0	2	2	2	5	2	3	4	1	1	1	1	4
9	1	1	2	2	2	2	2	2	2	0	3	3	2	2	1	1	1	1	1
10	1	1	2	0	0	1	0	2	2	2	5	4	2	2	2	2	2	2	9
11	1	2	1	2	0	2	2	2	2	0	5	3	6	4	1	1	1	6	10
12	1	0	1	0	0	2	0	2	2	1	4	4	7	2	3	2	2	5	4